

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/817,288	03/27/2001	Yoshitake Shinkai	826.1718	7742

21171 7590 07/15/2004

STAAS & HALSEY LLP
SUITE 700
1201 NEW YORK AVENUE, N.W.
WASHINGTON, DC 20005

EXAMINER

LESNIEWSKI, VICTOR D

ART UNIT	PAPER NUMBER
----------	--------------

2155

DATE MAILED: 07/15/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

09/817,288

Applicant(s)

SHINKAI ET AL.

Examiner

Victor Lesniewski

Art Unit

2155

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 27 March 2001.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-29 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-29 is/are rejected.
- 7) ☒ Claim(s) 16 is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☒ All b) ☐ Some * c) ☐ None of:
- 1) ☒ Certified copies of the priority documents have been received.
 - 2) ☐ Certified copies of the priority documents have been received in Application No. _____.
 - 3) ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date 4/16/2001.
- 4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____.
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: _____.

DETAILED ACTION

1. This application has been examined.
2. Claims 1-29 are now pending.

Priority

3. Receipt is acknowledged of papers submitted under 35 U.S.C. 119(a)-(d), which papers have been placed of record in the file.

Information Disclosure Statement

4. The IDS filed on 4/16/2001 has been considered.

Claim Objections

5. Claim 16 is objected to under 37 CFR 1.75(c), as being of improper dependent form for failing to further limit the subject matter of a previous claim. Applicant is required to cancel the claim, amend the claim to place it in proper dependent form, or rewrite the claim in independent form. In this case, claim 16 exactly repeats the limitations of claim 15 on which it depends, thereby failing to further limit claim 15.

Claim Rejections - 35 USC § 112

6. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

Art Unit: 2155

7. Claims 23-25 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

8. Concerning claim 23, there is a period present in the middle of the claim making it unclear where the claim actually ends. This makes the claim indefinite as it is unclear what exactly the applicant is claiming.

9. Claims 24 and 25 are rejected due to their dependence on claim 23.

Claim Rejections - 35 USC § 102

10. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

11. Claims 1-29 are rejected under 35 U.S.C. 102(b) as being clearly anticipated by the applicant's admitted prior art, namely the publication "Asynchronous File System Replication With a Strong Consistency" by Shinkai et al. dating to June 26, 1998.

12. The above stated publication has disclosed:

- <Claim 1>

A file replication system having plurality nodes connected to a network, shared files being distributed to the nodes, wherein a first node of the nodes comprises: a first token managing portion asking a second node of the nodes an access permission for a shared file when an access request takes place in first node (Section 2, especially *daemon acting*

as a token server and Section 1, especially *write and read requests from other nodes to the common file are forwarded to it*), and an IO request intercepting portion accepting an access to shared file, the access taking place the first node itself, asking said first token managing portion to acquire the access permission against the access request, and asking a node that has an update permission for the shared file to access to the shared file when said first token managing portion is not capable of acquiring the access permission (Section 2, especially *the filter intercepts an I/O request to the underlying file system from an application program and propagates it to other nodes asynchronously*), and a second node comprises a second token managing portion notifying a node that requests an access permission for a shared file of a node that has an update permission for the shared file as a response message when another node has an update permission for the shared file (Section 2, especially *the daemon sitting on every node receives an update notification associated with a write request*).

- <Claim 2>

A node, connected to another node through a network, having a file shared with a node, comprising: a token managing portion managing an access request for a shared file (Section 2, especially *daemon acting as a token server*); and an IO request intercepting portion asking said token managing portion to acquire an access permission for the shared file against an access request to the shared file in a node itself (Section 2, especially *the filter intercepts an I/O request to the underlying file system from an application program and propagates it to other nodes asynchronously*), wherein said token managing portion notifies said IO request intercepting portion of a node that has an

update permission in response to the access request of said IO request intercepting portion, and said IO request intercepting portion asks said node that has the update permission to access the shared file when said IO request intercepting portion is not capable of acquiring the access permission (Section 1, especially *write and read requests from other nodes to the common file are forwarded to it*).

- <Claim 3>

The node according to claim 2, further comprising: a system structure managing portion performing a restoration process of data of a shared file of the node itself when it is newly joined to a system, wherein while said system structure managing portion is restoring the shared file, when an access request for the shared file takes place in the node itself, said IO request intercepting portion asks another node that shares the shared file to access the shared file (Section 3.1, especially *read/write requests for the file is passed to the node holding the associated write token instead of revoking the write token when update potential remains on the owner node*).

- <Claim 4>

The node according to claim 2, further comprising: a changed data notifying portion propagating an updated content of the shared file to other node along with information that represents a dependent relationship with another update (Section 3.1.2, especially description of Dvector); and a received data processing portion reflecting the updated content to the shared file while assuring an order of the update based on the dependency relationship (Section 3.1.2, especially description of Rmatrix).

- <Claim 5>

The node according to claim 4, further comprising: a system state information portion storing information about propagation mode of an updated content for each of at least one shared file, wherein said changed data notifying portion propagates the update content based on information queued in said system information portion (Section 3.1.3, especially paragraph 1).

- <Claim 6>

The node according to claim 5, wherein the propagation mode is one of a synchronous mode in which it is assured that the updated content is propagated to all the nodes that share the shared file, a semi-synchronous mode in which it is assured that the updated content is propagated to the majority of nodes that share the shared file, and an asynchronous mode in which it is not acknowledged that the updated content is propagated to the nodes that share the shared file (Section 1, especially *the notification of updates to slaves is done asynchronously*).

- <Claim 7>

The node according to claim 4, wherein said system state information storing portion keeps information about each node that shares at least one shared file for each shared file (Section 3.1.2, especially description of Rmatrix).

- <Claim 8>

A node, connected to another node through a network, having file shared with node, comprising: a token managing portion asking another node to acquire an access permission for a shared file against an access request for the shared file in the node itself

(Section 2, especially *daemon acting as a token server* and Section 1, especially *write and read requests from other nodes to the common file are forwarded to it*); and an IO request intercepting portion accepting an access request for a shared file in the node itself, asking said token managing portion to acquire the access permission for the shared file against the access request, and asking a node that has an update permission for the shared file to access the shared file according to the access request when said token managing portion is not capable of acquiring the access permission for the shared file (Section 2, especially *the filter intercepts an I/O request to the underlying file system from an application program and propagates it to other nodes asynchronously*).

- <Claim 9>

A node, connected to another node through a network, having a file shared with a node, comprising: a permission request accepting portion accepting an access permission request of another node for a shared file (Section 2, especially *daemon acting as a token server*); and a token managing portion notifying first node that has issued the access permission request for a shared file of second node, when the second node has an update permission for the shared file (Section 2, especially *the daemon sitting on every node receives an update notification associated with a write request*).

- <Claim 10>

A file replication system having a plurality of nodes connected to a network, shared files being distributed to the nodes, wherein a first node the nodes comprises: first token managing means for asking a second node of the nodes for an access permission for a shared file when an access request takes place in the first node (Section 2, especially

daemon acting as a token server and Section 1, especially *write and read requests from other nodes to the common file are forwarded to it*), and IO request intercepting means for accepting access to a shared file, the access taking place in the first node itself, asking said first token managing means to acquire the access permission against the access request, and asking a node that has an update permission for the shared file to access to the shared file when said first token managing means is not capable of acquiring the access permission (Section 2, especially *the filter intercepts an I/O request to the underlying file system from an application program and propagates it to other nodes asynchronously*), and a second node comprises: second token managing means for notifying a node that requests an access permission for a shared file of a node that has an update permission for the shared file as a response message when another node has an update permission for the shared file (Section 2, especially *the daemon sitting on every node receives an update notification associated with a write request*).

- <Claim 11>

A node, connected to another node through a network, having a file shared with a node, comprising: token managing means for managing an access request for a shared file (Section 2, especially *daemon acting as a token server*); and IO request intercepting means for asking said token managing means to acquire an access permission for the shared file in response to an access request to the shared file in the node itself (Section 2, especially *the filter intercepts an I/O request to the underlying file system from an application program and propagates it to other nodes asynchronously*), wherein said token managing means notifies said IO request intercepting means of a node that has an

update permission in response to the access request of said IO request intercepting means, and said IO request intercepting means asks the node that has the update permission to access the shared file when said IO request intercepting means is not capable of acquiring the access permission (Section 1, especially *write and read requests from other nodes to the common file are forwarded to it*).

- <Claim 12>

A node, connected to another node through a network, having a file shared with the node, comprising: token managing means for asking another node to acquire an access permission for a shared file against an access request for the shared file in the node itself (Section 2, especially *daemon acting as a token server* and Section 1, especially *write and read requests from other nodes to the common file are forwarded to it*); and IO request intercepting means for accepting an access request for a shared file in the node itself, asking said token managing means to acquire the access permission for the shared file against the access request, and asking a node that has an update permission for the shared file to access the shared file according to the access request when said token managing means is not capable of acquiring the access permission for the shared file (Section 2, especially *the filter intercepts an I/O request to the underlying file system from an application program and propagates it to other nodes asynchronously*).

- <Claim 13>

A node, connected to another node through a network, having file shared with a node, comprising: permission request accepting means for accepting an access permission request of another node for shared file (Section 2, especially *daemon acting as a token*

server); and token managing means for notifying first node that has issued the access permission request for a shared file of second node, when the second node has an update permission for the shared file (Section 2, especially *the daemon sitting on every node receives an update notification associated with a write request*).

- <Claim 14>

A file replication control method for a system having a plurality of nodes connected to a network, each node sharing a file, comprising: causing an access requesting node to access a shared file of the access requesting node itself when the access requesting node has the latest data of a shared file (Section 2, especially *daemon acting as a token server*); and asking another node to access the shared file when said another node has the latest data (Section 1, especially *write and read requests from other nodes to the common file are forwarded to it*).

- <Claim 15>

The file replication control method according to claim 14, wherein said another node that has the update permission releases the update permission after an updated content that has a dependent relationship with an update performed at said another node itself, has been propagated to all the nodes (Section 1, especially *primary resigns only after all updated data was propagated to all slave nodes*).

- <Claim 16>

The file replication control method according to claim 15, wherein said another node that has the update permission to release the update permission after update that has a dependent relationship with the update performed at said another node itself, has been

propagated to all the nodes (Section 1, especially *primary resigns only after all updated data was propagated to all slave nodes*).

- <Claim 17>

The file replication control method according to claim 14, wherein said another node that has updated the shared file to asynchronously propagate an updated content to the other nodes (Section 1, especially *the notification of updates to slaves is done asynchronously*); and causing the node that updated the shared file to process an access request that takes place in another node while the updated content is being propagated (Section 3.1, especially *read/write requests for the file is passed to the node holding the associated write token instead of revoking the write token when update potential remains on the owner node*).

- <Claim 18>

The file replication control method according to claim 17, wherein the updated content is reflected in such a manner that order thereof is assured (Section 3.2.2, especially paragraph 1).

- <Claim 19>

The file replication control method according to claim 18, wherein a dependency information that represents order of other updates to be propagated to the other node along with the updated content (Section 3.1.2, especially paragraph 3).

- <Claim 20>

The file replication control method according to claim 19, wherein a node that has received the updated content to reflect the updated content on a shared file of the node

itself after receiving a previous updated content based on the dependency information (Section 3.1.3, especially paragraph 1).

- <Claim 21>

The file replication control method according to claim 14, wherein a propagation mode of an updated content is designated for each of at least one shared file (Section 1, especially paragraph 5).

- <Claim 22>

The file replication control method according to claim 14, wherein a node to which an updated content is propagated is designated for each of at least one shared file (Section 3.1.2, especially description of Wnumber).

- <Claim 23>

The file replication control method according to claim 14, further wherein restoring data of a shared file of a newly joined node; and operating a user program before data of the shared file is completely restored. a node performs a restoring process that restores data of a shared file belong to the node itself when the node is newly joined to a system, and operating a user program before the data of the shared file is completely restored (Section 3.1, especially paragraph 1).

- <Claim 24>

The file replication control method according to claim 23, wherein restored data is transmitted in such a manner that order of update requests for the shared file is assured (Section 3.1.2, especially paragraph 3).

- <Claim 25>

The file replication control method according to claim 23, wherein the node asks another node that shares the shared file to perform a process for an access request for the shared file when the access request takes place in the node itself before data is completely restored (Section 3.1, especially *read/write requests for the file is passed to the node holding the associated write token instead of revoking the write token when update potential remains on the owner node*).

- <Claim 26>

The file replication control method according to claim 14, wherein a node that has performed a systematic stop in which nodes that share a file are synchronously stopped to store a systematic stop state and the node synchronously resumes a process for the shared file without restoring data of the shared file (Section 4, especially reference to prior art coda file system).

- <Claim 27>

A file replication method for a system having a plurality of nodes connected to a network, comprising: causing a first node to request a token for accessing a file (Section 2, especially *daemon acting as a token server*); notifying the first node of a second node that has the token when the first node is not capable of acquiring the token (Section 2, especially *the daemon sitting on every node receives an update notification associated with a write request*); and causing the first node to ask the second node to access the file when the first node is notified that the first node is not capable of acquiring the token

(Section 2, especially *the filter intercepts an I/O request to the underlying file system from an application program and propagates it to other nodes asynchronously*).

- <Claim 28>

A computer-readable portable storage medium, when being used by a computer that composes a node connected to other node through a network, on which is recorded a program for causing the computer to execute a process, said process comprising: when the node accesses a shared file and a node itself has the latest data of the shared file, causing the node itself to access the shared file of the node itself (Section 2, especially *daemon acting as a token server*); and when another node has the latest data, causing the node itself to ask the node to access the shared file (Section 1, especially *write and read requests from other nodes to the common file are forwarded to it*).

- <Claim 29>

A computer-readable storage medium for storing a program that causes a computer that composes a node connected to another node through a network to perform the steps of: when a node issues an access request for a file shared with other node, judging whether or not a specific node has update permission for the shared file; and when the specific node has update permission, notifying the requesting node of the specific node that has the update permission (Section 2, especially *daemon acting as a token server*).

Since all the limitations of the invention as set forth in claims 1-29 were disclosed by this publication, claims 1-29 are rejected.

Conclusion

13. The prior art made of record and not relied upon is considered pertinent to the applicant's disclosure.

- Bogantz et al. (U.S. Patent Number 6,243,715) disclosed a replicated database synchronization method.
- Arnold et al. (U.S. Patent Number 6,263,360) disclosed methods for maintaining updated information on client/server object-oriented computed systems.
- Wahl et al. (U.S. Patent Number 6,324,654) disclosed a computer network remote data mirroring system.
- Straube et al. (U.S. Patent Number 6,412,017) disclosed a method for expediting the replication of at least one specified object to a replica in a distributed computer system.
- Zayas et al. (U.S. Patent Number 6,477,583) disclosed an infrastructure for supporting file replications.
- Dadiomov et al. (U.S. Patent Number 6,529,932) disclosed a system for distributed transaction processing with asynchronous message delivery.

14. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Victor Lesniewski whose telephone number is 703-308-6165.

The examiner can normally be reached on Monday through Thursday.


If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Hosain Alam can be reached on 703-308-6662. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Art Unit: 2155

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).



Victor Lesniewski
Patent Examiner
Group Art Unit 2155



HOSAIN ALAM
SUPERVISORY PATENT EXAMINER